S&H Form: PTO/SB/05 (9/99)

UTILITY PATENT APPLICATION TRANSMITTAL *(Only for new nonprovisional applications under 37 CFR 1.53(b))*	*Attorney Docket No.* 1293.1128/MJB *First Named Inventor or Application Identifier:* Jae-seong SHIM et al. *Express Mail Label No.*

JC760 U.S. PTO 07/10/00

JC598 U.S. PTO 09/612971 07/10/00

APPLICATION ELEMENTS *See MPEP chapter 600 concerning utility patent application contents.*	*ADDRESS TO:* **Assistant Commissioner for Patents** **Box Patent Application** **Washington, DC 20231**

1. [X] Fee Transmittal Form
2. [X] Specification, Claims & Abstract ..... [Total Pages: 16]
3. [X] Drawing(s) *(35 USC 113)* .... [Total Sheets: 12]
4. [ ] Oath or Declaration...................... [Total Pages: __]
   a. [ ] Newly executed (original or copy)
   b. [ ] Copy from a prior application (37 CFR 1.63(d)) *(for continuation/divisional with Box 17 completed)*
      i. [ ] DELETION OF INVENTOR(S)
      Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).
5. [ ] Incorporation by Reference (usable if Box 4b is checked)
   The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.
6. [ ] Microfiche Computer Program *(Appendix)*
7. [ ] Nucleotide and/or Amino Acid Sequence Submission *(if applicable, all necessary)*
   a. [ ] Computer Readable Copy
   b. [ ] Paper Copy (identical to computer copy)
   c. [ ] Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

8. [ ] Assignment Papers (cover sheet & document(s))
9. [ ] 37 CFR 3.73(b) Statement *(when there is an assignee)* [ ] Power of Attorney
10. [ ] English Translation Document *(if applicable)*
11. [ ] Information Disclosure Statement (IDS)/PTO-1449[ ] Copies of IDS Citations
12. [ ] Preliminary Amendment
13. [X] Return Receipt Postcard (MPEP 503) *(Should be specifically itemized)*
14. [ ] Small Entity Statement(s) [ ] Statement filed in prior application, status still proper and desired.
15. [X] Certified Copy of Priority Document(s) *(if foreign priority is claimed)*
16. [X] Other: New Application Fee Transmittal

17. If a CONTINUING APPLICATION, *check appropriate box and supply the requisite information:*

[ ] Continuation [ ] Divisional [ ] Continuation-in-part (CIP) of prior application No: ___/_______

18. CORRESPONDENCE ADDRESS

STAAS & HALSEY LLP
Attn: Michael D. Stein
700 Eleventh Street, N.W., Suite 500
Washington, DC 20001

Telephone: (202) 434-1500
Facsimile: (202) 434-1501

TITLE OF THE INVENTION

ERROR CORRECTION METHOD FOR HIGH DENSITY DISC

CROSS-REFERENCE TO RELATED APPLICATIONS

This application claims the benefit of Korean Application No. 99-27453, filed July 8, 1999, in the Korean Patent Office, the disclosure of which is incorporated herein by reference.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to an error correction method for optical discs, and more particularly, to an error correction method appropriate for high density discs.

2. Description of the Related Art

There are currently a variety of optical discs available, including a compact disc (CD), a digital versatile disc (DVD), and a high density DVD (HD-DVD), which requires higher density recording and reproducing than a DVD, and is currently under development. While a conventional DVD has a storage capacity of 4.7 GB, the HD-DVD has a storage capacity of 15 GB or more. The higher storage capacity of the HD-DVD is implemented by reducing the diameter of a beam spot for data recording/reproducing and increasing the line density.

The amount of data affected by a defect in an HD-DVD is far greater than the amount of data affected by the same length defect in a conventional DVD. Therefore, an HD-DVD requires stronger error correction than a conventional DVD.

FIG. 1 shows the structure of an error correction code (ECC) block in a conventional DVD. The error correction code block shown in FIG. 1 has a 10-byte parity for error correction of 172 bytes of data in the row direction, as an inner parity

(PI), and a 16-byte parity for error correction of 192 bytes of data in the column direction, as an outer parity (PO). Here, the capability of error correction by the PI is a maximum of 5 bytes, and that of the PO is a maximum of 16 bytes for erasure correction.

Assuming that an HD-DVD uses the same error correction method as a conventional DVD, the effect of a defect will now be explained in detail.

FIG. 2 illustrates the relationship of a beam spot and an object lens in an optical disc.

Table 1 illustrates the relationships among t, the thickness of a disc, NA, the numerical aperture of an object lens, 2R, the diameter of a beam spot, and k, the length of a defect.

Table 1

t (mm)	NA	R (mm)	2R (mm)	Remark	k, length of defect
0.6	0.6	0.248	0.496	DVD	k + 2R
	0.65	0.273	0.546		
0.3	0.65	0.136	0.272		
	0.85	0.193	0.286		
0.2	0.85	0.129	0.258		
0.1	0.7	0.049	0.098	DVD/3.88	
	0.85	0.064	0.128		

1) The effect of a large defect

Here, a large defect means a burst error which cannot be corrected by a PI, and is generated by a scratch, a finger print, a black dot, etc.

A defect which spans 5 bytes or more is a burst error which cannot be corrected by a PI. At this time, the length of a defect is k = 5 bytes x 16 channel bits x 0.133 μm (the length of 1 channel bit) = 10.64 μm.

When a 20 GB HD-DVD is compared to a 4.7 GB DVD, the line density increase is $(20/4.7)^{1/2}$. Accordingly, the same length defect damages 2.1 times more data in an HD-DVD than in a DVD.

Though an HD-DVD seems to be more advantageous than a DVD due to the HD-DVD's smaller spot size, the stabilization time required for restoring a reproduction signal (RF) in an HD-DVD is longer. Therefore, the effect of a spot size is thought to be similar in an HD-DVD and a DVD.

2) The effect of a small defect

Here, a small defect means a burst error which can be corrected by a PI, and is generated by dust and the like. The length of the defect is equal to or less than 10.64 μm.

According to table 1, when NA=0.85 and t=0.1mm, the diameter of a beam spot incident upon the surface of an HD-DVD is 0.128 μm, which is 1/3.88 times that of a DVD with a diameter of the beam spot being 0.496 μm. Therefore, the HD-DVD's probability of error occurrence by a small defect becomes 3.88 times greater than that of a DVD.

In addition, since the line density of an HD-DVD is 2.1 times greater than that of a conventional DVD, the probability of error in an HD-DVD is 8.148 times (3.88 x 2.1 = 8.148) greater than that of a DVD for the same size defect. This means that when an HD-DVD uses the same modulation method as a DVD, error correction by a PI must be available for about 40.74 bytes (5 bytes x 8.148). Therefore, an HD-DVD requires a great number of PIs.

In the previous DVD error correction method shown in FIG. 1, in order to raise the burst error correction capability, the number of data columns must be increased in the PI direction, while the number of data rows must be decreased in the PO direction.

However, when n, the number of data columns in the PI direction, exceeds 256, a Galois Field operation GF(28) cannot be performed.

Thus, the previous error correction method in a DVD as shown in FIG. 1 cannot be easily applied to HD-DVD.

SUMMARY OF THE INVENTION

Accordingly, it is an object of the present invention to provide an error correction method appropriate for an HD-DVD.

It is another object to provide a basic addressing structure appropriate for the HD-DVD.

Additional objects and advantages of the invention will be set forth in part in the description which follows, and, in part, will be obvious from the description, or may be learned by practice of the invention.

To accomplish the above objects of the present invention, there is provided an error correction method adding inner parity and outer parity to an error correction block having a size of n bytes x m x o, the error correction method having the steps of obtaining a plurality of inner parity blocks (PI blocks) by segmenting the error correction block in an inner parity (PI) direction into x segments (here, x is an integer equal to or greater than 2); generating e-byte PI for each of the plurality of PI blocks generated by segmenting, and adding the PIs in the PI direction; and generating f-byte outer parity (PO) in the PO direction of the error correction block having PIs, and adding the POs in a PO direction.

It is preferable that the data frame, which forms an error correction block, is formed with two 2-KB user data blocks.

Also, it is preferable that the data frame has EDCs for correcting errors in user data.

BRIEF DESCRIPTION OF THE DRAWINGS

invention will become more apparent by describing in detail a preferred embodiment thereof with reference to the attached drawings in which:]

These and other objects and advantages of the invention will become apparent and more readily appreciated from the following description of the preferred embodiments, taken in conjunction with the accompanying drawings of which:

FIG. 1 is the structure of an error correction code (ECC) block in a conventional digital versatile disc (DVD);

FIG. 2 illustrates the relationship between a beam spot and an object lens in an optical disc;

FIG. 3 illustrates the relationships between an ECC block, an inner parity and an outer parity in the error correction method according to an embodiment of the present invention;

FIG. 4 illustrates the effect by an interleave between inner parity (PI) blocks in the same row;

FIG. 5 shows the process for performing an error correction method according to the embodiment of the present invention;

FIG. 6 illustrates the structure of a data frame after it has been scrambled in the error correction method of FIG. 5;

FIGS. 7A and 7B illustrate generation of inner parity and outer parity in an error correction block in the error correction method of FIG. 5;

FIGS. 8A and 8B illustrate the result of interleaving to the inner parity direction in the error correction method of FIG. 5;

FIG. 9 illustrates the result of interleaving the result shown in FIG. 8 again in the inner parity direction; and

FIGS. 10A through 10D illustrate the result of interleaving in the outer parity direction in the error correction method in FIG. 5.

DESCRIPTION OF THE PREFERRED EMBODIMENT

Reference will now be made in detail to the present preferred embodiment of the present invention, examples of which are illustrated in the accompanying drawings, wherein like reference numerals refer to like elements throughout.

FIG. 3 illustrates the relationships between an error correction block, an inner parity (PI) and an outer parity (PO) in an error correction method according to an embodiment of the present invention. As a method for improving a burst error

correction capability in using the same number of parities, it is preferable that the number of data columns is increased in the PI direction and the number of data rows is decreased in the PO direction.

However, since a Galois Field operation cannot be performed when n, the number of data columns in the PI direction exceeds 256, the present invention uses a multi-way PI error correction method.

That is, n, the number of data columns in a row is divided into segments with an appropriate size (x), and then, an e-byte PI is added to each segmented PI block. Here, the size n/x is determined to be an appropriate size for adding a synchronous signal, and n, x, and e are determined so that n/x + e is less than or equal to 256.

If the number of data frames in the PO direction is 16, m (rows) x 16 + f (rows) is less than or equal to 256. Furthermore, x (the number of PI direction segments), and f, (the number of PO direction parities), are decided so that the result of multiplication of x with f is divided by o, the number of data frames, without a remainder. In this case, f can be not equal to o, the number of data frames, unlike a conventional DVD, in which f is equal to o.

The error correction block shown in FIG. 3 causes a problem when the block is recorded on a disc immediately after channel-modulation. That is, when a small defect occurs and e/2 bytes of data or more are damaged, correction by a PI becomes impossible. Therefore, after adding a flag indicating that all data in the corresponding PI block is not corrected, the data must be sent to an error correction process by a PO. When greater than or equal to f data is sent to a PO with a flag indicating that the data is not corrected by a PI, the PO cannot correct the error either.

In order to effectively correct small defects and sporadically occurring errors, interleaving is performed in the PI direction in x PI blocks.

FIG. 4 illustrates the effect of interleaving PI blocks in the same row. As shown in FIG. 4, even though a burst error occurs, the burst error changes into sporadic errors due to the interleaving between the PI blocks. Therefore, even when

e/2 or more bytes of data are damaged, the number of errors are reduced to equal to or less than e/2 in a PI block after interleaving, and error correction becomes possible.

There is another method in which e-byte parity is added to each x-th data in the same PI direction. In one method, interleaving is performed among PI blocks in different rows in order to increase the interleaving effect. In this method, however, there is greater delay between the time when error correction is completed and the time when data is output. Therefore, it is preferable that the scope of interleaving is determined as a function of the delay and the size of burst defects to be corrected.

FIG. 5 shows the process for performing an error correction method according to the embodiment of the present invention.

First, data for detection (IED) is added to address information (ID) 502 to yield "ID + IED" 504.

Next, reserve space (RSV) for storing future scalability, user information, producer information, copyright protection, etc., and 4KB user data is added To "ID + IED" 504 to yield "(ID + IED) & RSV & 4KB USER DATA" 506.

Next, 4 KB of user data is divided into 2KB, considering compatibility to an existing compact disc (CD) and a digital versatile disc (DVD), and then, an error detection code (EDC) for detecting an error is added. By doing so, one data frame 508 is formed.

Next, in order to obtain data protection, channel modulation, and servo capacity, scrambling is performed on data frame 508. For example, in order to properly perform scrambling of data on a 20 GB-level HD-DVD having 4KB data frames and a 64 KB basic unit for error correction, the length of the cycle of the random data generator in an HD-DVD having a 64 KB basic unit for error correction and a 4 KB user data in one data frame is designed to be 64 K, which is advantageous in suppressing direct current (DC) component during servo operation and modulation.

FIG. 6 illustrates the structure of the data frame 510 after it has been scrambled in the error correction method in FIG. 5. Referring to the example in FIG. 6, data frame 510 is formed with a 4-byte ID, a 2-byte IED, an 18-byte RSV, two 2-KByte

user data blocks, and two 4-byte EDCs. Here, one data frame 510 is 688 bytes in the PI direction (column direction), and 6 rows in the PO direction (row direction).

Returning now to FIG. 5, one error correction block 512 is formed by gathering 16 data frames 510 shown, and a PI and a PO are added to the block. This error correction block 512 then undergoes the steps of PI/PO encoding and PI/PO interleaving to form recording block 514. Finally, a synchronous signal is added to recording block 514 yielding physical block 516, which is then recorded on a disc.

We now turn to FIGS. 7-10, which illustrate in detail the PI/PO encoding and interleaving of the present invention.

FIGS. 7A and 7B illustrate the generation of inner parity and outer parity in an error correction block shown as "4 way PI ENCODING \ SINGLE PO ENCODING" in FIG. 5. Referring to FIGS. 7A and 7B, 16 data frames 510 are lined up and then, four PIs, each of which have 8 bytes in the PI direction, are added, and a PO, which has 12 byte to the PO direction, is added.

1) PO

PO is generated by using the Reed Solomon code RS(108, 96, 13).

That is, for data (B0,0 ~ Bi,j, i=0~95, j=0~687), B96,0 ~ Bi,j are generated.

2) PI

PI is generated by using RS(180, 172, 9).

That is, for data (Bi,0 ~ Bi,171, i=0~107), Bi,688 ~ Bi,695 (i=0 ~ 107) are generated; for data (Bi,172 ~ Bi,343, i=0~107), Bi,696 ~ Bi,703 (i=0 ~ 107) are generated; for data (Bi,344 ~ Bi,545, i=0~107), Bi,704 ~ Bi,711 (i=0 ~ 107) are generated; and for data (Bi,546 ~ Bi,687, i=0~107), Bi,712 ~ Bi,719 (i=0 ~ 107) are generated.

Parties are generated in 4 ways in the PI direction so that no PI correction unit (including parity) exceeds 256, thus a GF (28) operation in a Galois Field can be performed. This also permits the addition of correction incapability flags in four divided units for better erase correction in the PO correction process.

Furthermore, interleaving four PI blocks improves PI correction capability. In the present invention, such an error correction method is referred to as Reed-Solomon multiple way PI or PO product code (RS-MWPC).

After PI/PO encoding, a burst error in the PI direction is changed into sporadic errors, and in order to protect PI and PO, interleaving is performed in the PI direction. FIGS. 8A and 8B illustrate the result of this interleaving process, which is shown in FIG. 5 as "DATA INTERLEAVE COLUMN INTERLEAVE OF PI". Referring to FIGS. 8A and 8B, data in four PI blocks is reallocated one by one in a predetermined turn in the data section and the parity section.

FIG. 9 illustrates the result of interleaving the result shown in FIGS. 8A and 8B again in the PI direction. PI divides each 8 bytes in the PI direction and performs interleaving. This is to prevent occurrence of burst errors in PIs.

When interleaving in PIs is completed, 12 rows including PO + PI parities from the 97^{th} row to the 108^{th} row are reorganized into 16 rows. The reason why 12 rows including PO + PI parities can be reorganized into 16 rows is that the result of multiplication of 4 (x), which is the number of PI direction segments, by 12 (f), which is the number of PO + PI parity rows, is 16 (o), the number of data frames. To achieve this, 720 bytes (688 + 32) in the first PO + PI parity row is multiplied by 3/4, and then, 540 bytes become the first new PO + PI parity row, and the remaining 720-540=180 bytes are passed to the second PO + PI parity row. The 180 bytes are added to 720 bytes that are in the second PO + PI parity row, and then the first 540 bytes in the result of the addition are changed into the second PO + PI parity row.

By doing so, the 12 rows are changed into a total of 16 rows of new PO + PI parity rows. By interleaving to the PO direction from the first row, all interleaving is finished and a total of 16 recording frames are reorganized as shown in FIGS. 10A and 10B. After inserting a synchronous signal and performing channel modulation, this data is in a form that can be actually recorded on optical disc.

As described above, the error correction method according to the present invention enhances error correction capability in an HD-DVD while maintaining redundancy of parity code on a level similar to conventional DVDs.

Although a few preferred embodiments of the present invention have been shown and described, it would be appreciated by those skilled in the art that changes may be made in these embodiments without departing from the principles and spirit of the invention, the scope of which is defined in the claims and their equivalents.

What is claimed is:

1. An error correction method adding an inner parity of e bytes and an outer parity of f bytes to an error correction block having a size of n bytes (in a row direction) x m bytes (in a column direction), the error correction method comprising:

obtaining a plurality of inner parity blocks (PI blocks) by segmenting the error correction block in an inner parity (PI) direction into x segments (here, x is an integer equal to or greater than 2);

generating e-byte PI for each of the plurality of PI blocks generated by segmenting, and adding the PIs in the PI direction; and

generating f-byte outer parity (PO) in a PO direction of the error correction block having PIs, and adding the POs in the PO direction.

2. The error correction method of claim 1, wherein the PIs are Reed-Solomon signs and satisfy $(n/x) + e \geq 256$.

3. The error correction method of claim 2, wherein $(n+e)$ x $(m+f)$ is less than or equal to 64K.

4. The error correction method of claim 3, wherein n is 688 and m is 96.

5. The error correction method of claim 4, wherein x is 172 and e is 8.

6. The error correction method of claim 5, wherein f is 12.

7. The error correction method of claim 1, further comprising:

interleaving a plurality of data groups and the plurality of PIs in the PI direction in the error correction blocks having PIs and POs.

8. The error correction method of claim 7, wherein the interleaving further comprises:

gathering bytes having the same order in each of the data groups; and

allocating the gathered bytes sequentially according to their order.

9. The error correction method of claim 8, wherein the reallocating is performed in the PI groups in a single data row.

10. The error correction method of claim 7, wherein the reallocating further comprises reallocating a plurality of PIs (PI0, PI1, ..., PIn/x) by gathering bytes having a same order in bytes included in each of the plurality of Pis, thereby forming reallocated Pis groups.

11. The error correction method of claim 10, wherein the reallocating is performed in the PIs in a single data row.

12. The error correction method of claim 10, further comprising:

moving and allocating the reallocated PIs between the reallocated PIs groups.

13. The error correction method of claim 11, further comprising:

interleaving the POs in the PO direction.

14. The error correction method of claim 13, wherein the PO direction interleaving further comprises:

obtaining an n x f byte bit stream by lining up the f-byte POs sequentially, and forming a divided PO by dividing the bit stream into each {(n x f)/m}; and

moving and allocating the divided PO in the PO direction in each row.

15. The error correction method of claim 4, wherein n x m is a basic address unit recorded on a disk, the method further comprising:

forming a data frame with a 4-byte ID, a 2-byte IED, an 18-byte RSV, two 2-KB user data blocks, and two 4-byte EDCs.

16. The error correction method of claim 1, further comprising determining f, which is a number of PO direction parities, and x, which is a number of PI direction segments, are decided so that a result of multiplication of x with f can be divided by o, which is a number of data frames in one error correction block, without remainder, and a recording frame is formable even when f is not equal to o.

17. The error correction method of claim 16, wherein a GF (28) operation in a Galois Field can be performed.

18. The error correction method of claim 8, wherein the reallocating is performed in the PI groups in a plurality of data rows.

19. An error correction method directed to an error correction block having data an inner parity direction and an outer parity direction, comprising:

segmenting the error correction block in the inner parity direction to form a plurality of inner parity segments.

20. The error correction method of claim 19, further comprising:

generating an e-byte inner parity for each of the plurality of inner parity segments; and

adding the e-byte inner parities to form a plurality of inner parity blocks.

21. The error correction method of claim 20, further comprising:

generating an f-byte outer parity; and

adding the f-byte outer parities in the outer parity direction.

22. The error correction method of claim 21, further comprising adding the e-byte inner parities to the inner parity segments in the inner parity direction.

23. The error correction method of claim 22, further comprising interleaving the data after adding the e-byte parities to the inner parity segments.

24. The error correction method of claim 23, wherein the interleaving of the data comprises interleaving in the inner parity direction.

25. The error correction method of claim 24; wherein the interleaving of the data in the inner parity direction comprises interleaving the data within the inner parity blocks.

26. The error correction method of claim 25, wherein the interleaving of the data in the inner parity direction comprises interleaving four inner parity blocks one by one in a predetermined turn.

27. The error correction method of claim 26, wherein the interleaving of the data comprises interleaving the data in the outer parity direction.

28. The error correction method of claim 27, wherein the interleaving of the data comprises interleaving a quantity of the data in relation to the size of a burst error.

29. An optical disk comprising:

an error correction block, comprising:

a plurality of inner parity blocks, each said inner parity block comprising an e-byte inner parity in an inner parity direction; and

a plurality of f-byte outer parities in an outer parity direction.

30. The optical disk of claim 29, further comprising a plurality of data groups interleaved with the inner parity blocks.

31. The optical disk of claim 30, wherein the plurality of f-byte outer parities are interleaved in the outer parity direction.

32. The optical disk of claim 31, wherein the optical disk is a digital versatile disk (DVD).

33. The optical disk of claim 32, wherein the digital versatile disk is a high density digital versatile disk (HD-DVD).

34. The optical disk of claim 33, wherein the high density digital versatile disk has a storage capacity of at least 15 GB.

ABSTRACT

An error correction method for optical discs, and more particularly, an error correction method appropriate to high density discs is provided. The error correction method adds inner parity and outer parity to an error correction block of size n byte x m x o. The method comprises the steps of obtaining a plurality of inner parity blocks (PI blocks) by segmenting the error correction block in the inner parity (PI) direction into x segments; generating e-byte PI for each of the plurality of PI blocks generated by segmenting, and adding the e-bytes to the PI blocks PIs to the PI direction; and generating f-byte outer parity (PO) in the PO direction of the error correction block, and adding the POs to the PO direction. The error correction method enhances error correction capability while maintaining a redundancy of parity signal on a level similar to conventional DVDs.

PRIOR ART

FIG. 1

172 BYTES | PI (10 BYTES)

192 BYTES | PO(16BYTES)

$B_{0,0}$	$B_{0,1}$	• • •	$B_{0,170}$	$B_{0,171}$	$B_{0,172}$	• • •	$B_{0,181}$
$B_{1,0}$	$B_{1,1}$	• • •	$B_{1,170}$	$B_{1,171}$	$B_{1,172}$	• • •	$B_{1,181}$
$B_{2,0}$	$B_{2,1}$	• • •	$B_{2,170}$	$B_{2,171}$	$B_{2,172}$	• • •	$B_{2,181}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
$B_{189,0}$	$B_{189,1}$	• • •	$B_{189,170}$	$B_{189,171}$	$B_{189,172}$	• • •	$B_{189,181}$
$B_{190,0}$	$B_{190,1}$	• • •	$B_{190,170}$	$B_{190,171}$	$B_{190,172}$	• • •	$B_{190,181}$
$B_{191,0}$	$B_{191,1}$	• • •	$B_{191,170}$	$B_{191,171}$	$B_{191,172}$	• • •	$B_{191,181}$
$B_{192,0}$	$B_{192,1}$	• • •	$B_{192,170}$	$B_{192,170}$	$B_{192,172}$	• • •	$B_{192,181}$
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •
$B_{207,0}$	$B_{207,1}$	• • •	$B_{207,170}$	$B_{207,171}$	$B_{207,172}$	• • •	$B_{207,181}$

PRIOR ART

FIG. 2

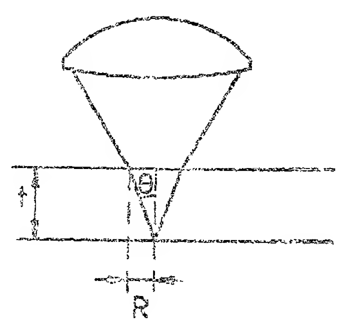

FIG. 3

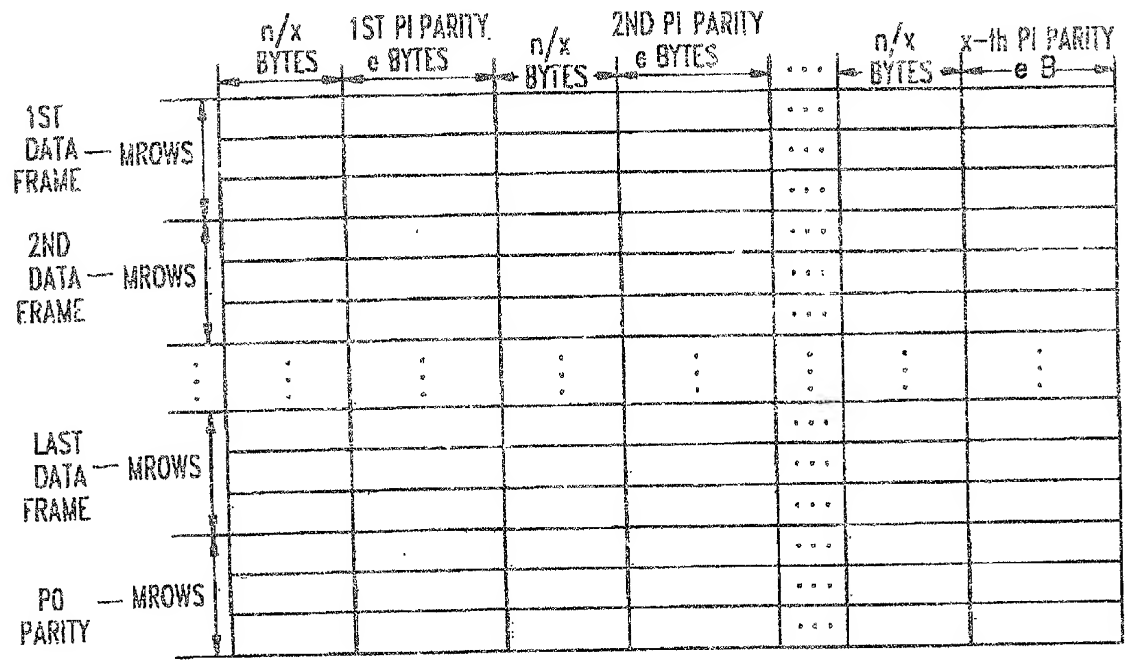
n/x BYTES
1ST PI PARITY e BYTES
n/x BYTES
2ND PI PARITY e BYTES
n/x BYTES
x-th PI PARITY e B
1ST DATA FRAME — MROWS
2ND DATA FRAME — MROWS
LAST DATA FRAME — MROWS
PO PARITY — MROWS

FIG. 4

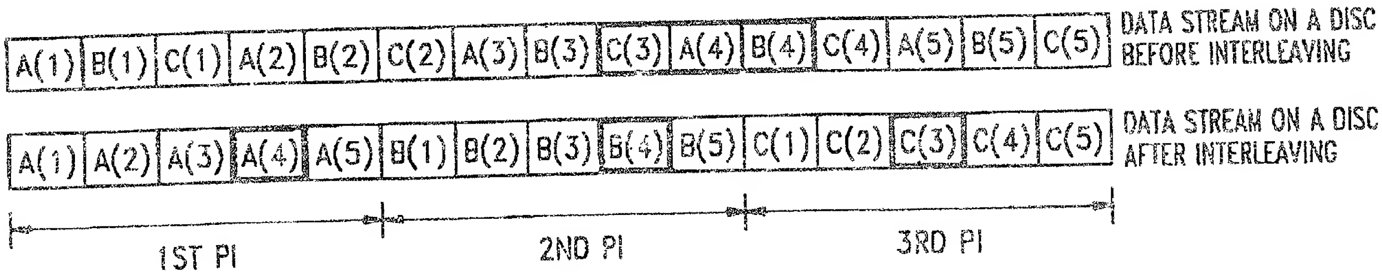
A(1) B(1) C(1) A(2) B(2) C(2) A(3) B(3) C(3) A(4) B(4) C(4) A(5) B(5) C(5)
DATA STREAM ON A DISC BEFORE INTERLEAVING
A(1) A(2) A(3) A(4) A(5) B(1) B(2) B(3) B(4) B(5) C(1) C(2) C(3) C(4) C(5)
DATA STREAM ON A DISC AFTER INTERLEAVING
1ST PI
2ND PI
3RD PI

FIG. 5

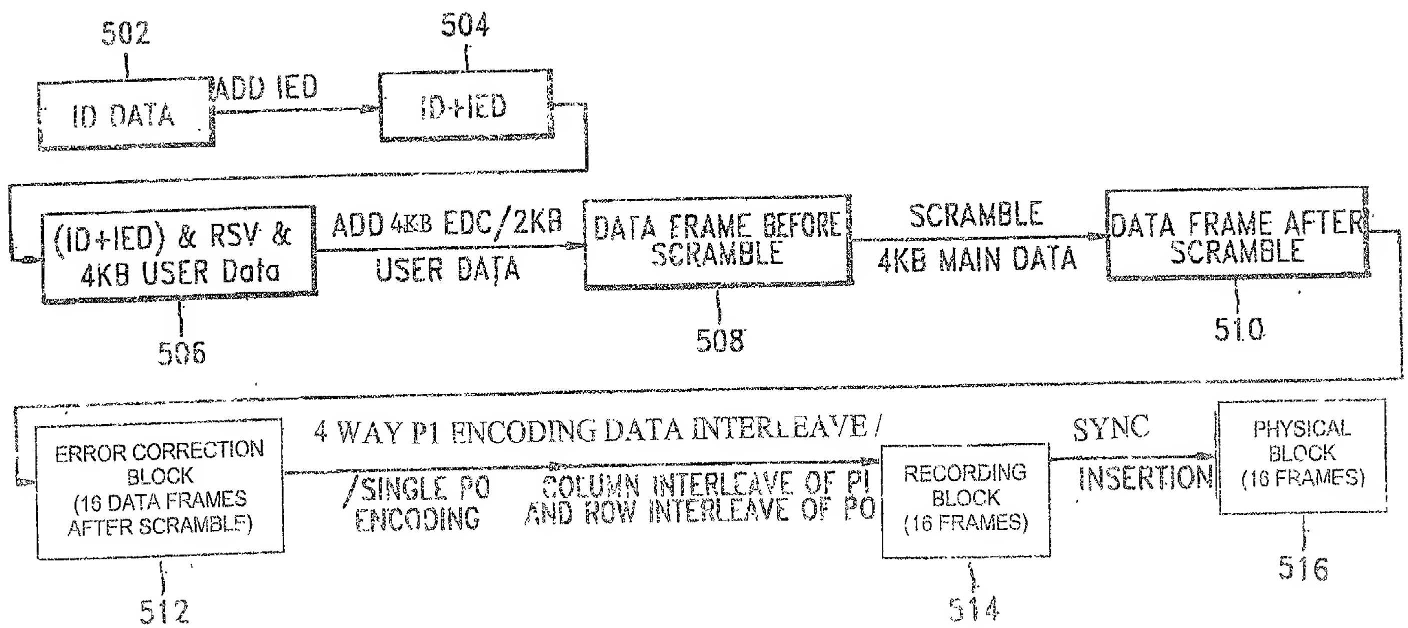
502
ID DATA
ADD IED
504
ID+IED
(ID+IED) & RSV & 4KB USER Data
506
ADD 4KB EDC/2KB USER DATA
DATA FRAME BEFORE SCRAMBLE
508
SCRAMBLE 4KB MAIN DATA
DATA FRAME AFTER SCRAMBLE
510
ERROR CORRECTION BLOCK (16 DATA FRAMES AFTER SCRAMBLE)
512
4 WAY P1 ENCODING DATA INTERLEAVE /
/SINGLE PO ENCODING
COLUMN INTERLEAVE OF PI AND ROW INTERLEAVE OF PO
RECORDING BLOCK (16 FRAMES)
514
SYNC INSERTION
PHYSICAL BLOCK (16 FRAMES)
516

FIG. 6

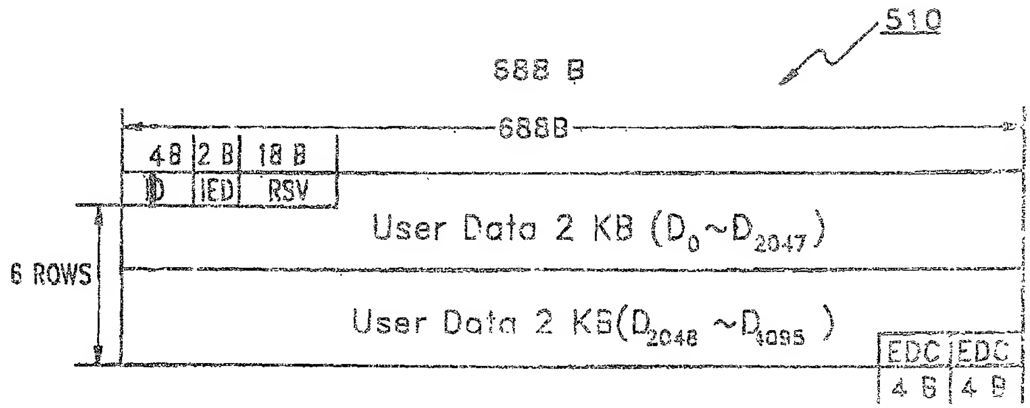
510
688 B
688B
4B
2 B
18 B
ID
IED
RSV
User Data 2 KB (D_0~D_{2047})
6 ROWS
User Data 2 KB(D_{2048} ~D_{4095})
EDC
EDC
4 B
4 B

FIG. 9

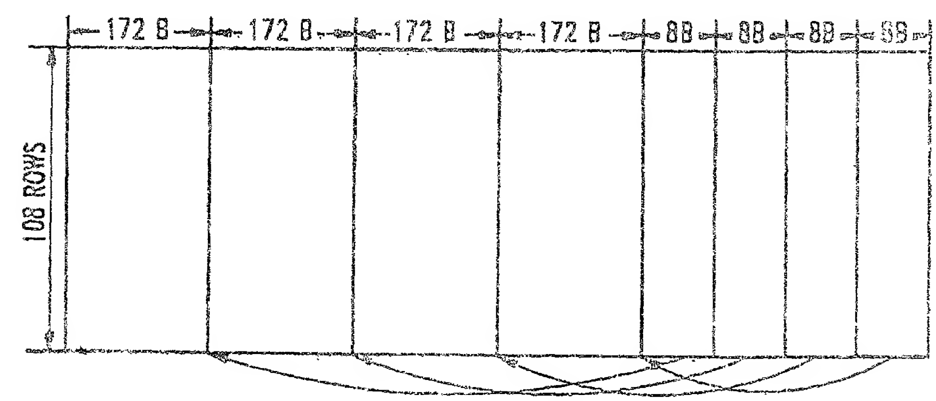
172 B
172 B
172 B
172 B
8B
8B
8B
8B
108 ROWS

FIG. 7A

	172B			172B			172B			172B		
16 DATA FRAMES 96 ROWS	$B_{0,0}$	...	$B_{0,171}$	$B_{0,172}$	...	$B_{0,343}$	$B_{0,344}$	...	$B_{0,515}$	$B_{0,516}$	...	$B_{0,687}$
	$B_{1,0}$	...	$B_{1,171}$	$B_{1,172}$	...	$B_{1,343}$	$B_{1,344}$	...	$B_{1,515}$	$B_{1,516}$	...	$B_{1,687}$
	$B_{2,0}$	...	$B_{2,171}$	$B_{2,172}$	...	$B_{2,343}$	$B_{2,344}$	...	$B_{2,515}$	$B_{2,516}$	...	$B_{2,687}$
	⋮	...	⋮	⋮	...	⋮	⋮	...	⋮	⋮	...	⋮
	$B_{93,0}$	...	$B_{93,171}$	$B_{93,172}$	...	$B_{93,343}$	$B_{93,344}$	...	$B_{93,515}$	$B_{93,516}$	...	$B_{93,687}$
	$B_{94,0}$	...	$B_{94,171}$	$B_{94,172}$	...	$B_{94,343}$	$B_{94,344}$	...	$B_{94,515}$	$B_{94,516}$	...	$B_{94,687}$
	$B_{95,0}$	...	$B_{95,171}$	$B_{95,172}$	...	$B_{95,343}$	$B_{95,344}$	...	$B_{95,515}$	$B_{95,516}$	...	$B_{95,687}$
PO 12 ROWS	$B_{96,0}$	...	$B_{96,171}$	$B_{96,172}$	...	$B_{96,343}$	$B_{96,344}$	...	$B_{96,515}$	$B_{96,516}$	...	$B_{96,687}$
	...	...	...	...	...	...	...	...	...	...	...	...
	$B_{107,0}$	...	$B_{107,171}$	$B_{107,172}$	...	$B_{107,343}$	$B_{107,344}$	...	$B_{107,515}$	$B_{107,516}$	...	$B_{107,687}$

FIG. 7B

1ST PI 8B			2ND PI 8B			3RD PI 8B			4TH PI 8B		
$B_{0,688}$	⋯	$B_{0,695}$	$B_{0,696}$	⋯	$B_{0,703}$	$B_{0,704}$	⋯	$B_{0,711}$	$B_{0,712}$	⋯	$B_{0,719}$
$B_{1,688}$	⋯	$B_{1,695}$	$B_{1,696}$	⋯	$B_{1,703}$	$B_{1,704}$	⋯	$B_{1,711}$	$B_{1,712}$	⋯	$B_{1,719}$
$B_{2,688}$	⋯	$B_{2,695}$	$B_{2,696}$	⋯	$B_{2,703}$	$B_{2,704}$	⋯	$B_{2,711}$	$B_{2,712}$	⋯	$B_{2,719}$
⋮	⋯	⋮	⋮	⋯	⋮	⋮	⋯	⋮	⋮	⋯	⋮
$B_{93,688}$	⋯	$B_{93,695}$	$B_{93,696}$	⋯	$B_{93,703}$	$B_{93,704}$	⋯	$B_{93,711}$	$B_{93,712}$	⋯	$B_{93,719}$
$B_{94,688}$	⋯	$B_{94,695}$	$B_{94,696}$	⋯	$B_{94,703}$	$B_{94,704}$	⋯	$B_{94,711}$	$B_{94,712}$	⋯	$B_{94,719}$
$B_{95,688}$	⋯	$B_{95,695}$	$B_{95,696}$	⋯	$B_{95,703}$	$B_{95,704}$	⋯	$B_{95,711}$	$B_{95,712}$	⋯	$B_{95,719}$
$B_{96,688}$	⋯	$B_{96,695}$	$B_{96,696}$	⋯	$B_{96,703}$	$B_{96,704}$	⋯	$B_{96,711}$	$B_{96,712}$	⋯	$B_{96,719}$
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
$B_{107,688}$	⋯	$B_{107,695}$	$B_{107,696}$	⋯	$B_{107,703}$	$B_{107,704}$	⋯	$B_{107,711}$	$B_{107,712}$	⋯	$B_{107,719}$

FIG. 8A

6888

96 ROWS / 12 ROWS

96 ROWS	$B_{0,0}$	$B_{0,172}$	$B_{0,344}$	$B_{0,516}$	$B_{0,1}$	···	$B_{0,171}$	$B_{0,343}$	$B_{0,515}$	$B_{0,687}$
	$B_{1,0}$	$B_{1,172}$	$B_{1,344}$	$B_{1,516}$	$B_{1,1}$	···	$B_{1,171}$	$B_{1,343}$	$B_{1,515}$	$B_{1,687}$
	$B_{2,0}$	$B_{2,172}$	$B_{2,344}$	$B_{2,516}$	$B_{2,1}$	···	$B_{2,171}$	$B_{2,343}$	$B_{2,515}$	$B_{2,687}$
	⋮	⋮	⋮	⋮	⋮	···	⋮	⋮	⋮	⋮
	$B_{93,0}$	$B_{93,172}$	$B_{93,344}$	$B_{93,516}$	$B_{93,1}$	···	$B_{93,171}$	$B_{93,343}$	$B_{93,515}$	$B_{93,687}$
	$B_{94,0}$	$B_{94,172}$	$B_{94,344}$	$B_{94,516}$	$B_{94,1}$	···	$B_{94,171}$	$B_{94,343}$	$B_{94,515}$	$B_{94,687}$
	$B_{95,0}$	$B_{95,172}$	$B_{95,344}$	$B_{95,516}$	$B_{95,1}$	···	$B_{95,171}$	$B_{95,343}$	$B_{95,515}$	$B_{95,687}$
12 ROWS	$B_{96,0}$	$B_{96,172}$	$B_{96,344}$	$B_{96,516}$	$B_{96,1}$	···	$B_{96,171}$	$B_{96,343}$	$B_{96,515}$	$B_{96,687}$
	···	···	···	···	···	···	···	···	···	···
	$B_{107,0}$	$B_{107,172}$	$B_{107,344}$	$B_{107,516}$	$B_{107,1}$	···	$B_{107,171}$	$B_{107,343}$	$B_{107,515}$	$B_{107,687}$

FIG. 8B

32B

$B_{0,688}$	$B_{0,696}$	$B_{0,704}$	$B_{0,712}$	$B_{0,689}$	···	$B_{0,711}$	$B_{0,719}$
$B_{1,688}$	$B_{1,696}$	$B_{1,704}$	$B_{1,712}$	$B_{1,689}$	···	$B_{1,711}$	$B_{1,719}$
$B_{2,688}$	$B_{2,696}$	$B_{2,704}$	$B_{2,712}$	$B_{2,689}$	···	$B_{2,711}$	$B_{2,719}$
⋮	⋮	⋮	⋮	⋮	···	⋮	⋮
$B_{93,688}$	$B_{93,696}$	$B_{93,704}$	$B_{93,712}$	$B_{93,689}$	···	$B_{93,711}$	$B_{93,719}$
$B_{94,688}$	$B_{94,696}$	$B_{94,704}$	$B_{94,712}$	$B_{94,689}$	···	$B_{94,711}$	$B_{94,719}$
$B_{95,688}$	$B_{95,696}$	$B_{95,704}$	$B_{95,712}$	$B_{95,689}$	···	$B_{95,711}$	$B_{95,719}$
$B_{95,688}$	$B_{96,696}$	$B_{96,704}$	$B_{96,712}$	$B_{96,689}$	···	$B_{96,711}$	$B_{96,719}$
···	···	···	···	···	···	···	···
$B_{107,688}$	$B_{107,696}$	$B_{107,704}$	$B_{107,712}$	$B_{107,689}$	···	$B_{107,711}$	$B_{107,719}$

FIG. 10A

RECORDING FRAME

172B					8B				
$B_{0,0}$	$B_{0,172}$	…	$B_{0,386}$	$B_{0,558}$	$B_{0,688}$	$B_{0,696}$	…	$B_{0,705}$	$B_{0,713}$
$B_{1,0}$	$B_{1,172}$	…	$B_{1,386}$	$B_{1,558}$	$B_{1,688}$	$B_{1,696}$	…	$B_{1,705}$	$B_{1,713}$
⋮	⋮	…	⋮	⋮	⋮	⋮	…	⋮	⋮
$B_{5,0}$	$B_{5,172}$	…	$B_{5,386}$	$B_{5,558}$	$B_{5,688}$	$B_{5,696}$	…	$B_{5,705}$	$B_{5,713}$
$B_{96,0}$	$B_{96,172}$	…	$B_{96,386}$	$B_{96,558}$	$B_{96,688}$	$B_{96,696}$	…	$B_{96,705}$	$B_{96,713}$
$B_{6,0}$	$B_{6,172}$	…	$B_{6,386}$	$B_{6,558}$	$B_{6,688}$	$B_{6,696}$	…	$B_{6,705}$	$B_{6,713}$
$B_{7,0}$	$B_{7,172}$	…	$B_{7,386}$	$B_{7,558}$	$B_{7,688}$	$B_{7,696}$	…	$B_{7,705}$	$B_{7,713}$
⋮	⋮	…	⋮	⋮	⋮	⋮	…	⋮	⋮
$B_{11,0}$	$B_{11,172}$	…	$B_{11,386}$	$B_{11,558}$	$B_{11,688}$	$B_{11,696}$	…	$B_{11,705}$	$B_{11,713}$
$B_{96,120}$	$B_{96,301}$	…	$B_{95,515}$	$B_{95,687}$	$B_{96,694}$	$B_{96,702}$	…	$B_{96,711}$	$B_{96,719}$
⋮	⋮	…	⋮	⋮	⋮	⋮	…	⋮	⋮
$B_{90,0}$	$B_{90,172}$	…	$B_{90,386}$	$B_{90,558}$	$B_{90,688}$	$B_{90,696}$	…	$B_{90,705}$	$B_{90,713}$
$B_{91,0}$	$B_{91,172}$	…	$B_{91,386}$	$B_{91,558}$	$B_{91,688}$	$B_{91,696}$	…	$B_{91,705}$	$B_{91,713}$
⋮	⋮	…	⋮	⋮	⋮	⋮	…	⋮	⋮
$B_{95,0}$	$B_{95,172}$	…	$B_{95,386}$	$B_{95,558}$	$B_{95,688}$	$B_{95,696}$	…	$B_{95,705}$	$B_{95,713}$
$B_{107,43}$	$B_{107,215}$	…	$B_{107,129}$	$B_{107,601}$	$B_{107,690}$	$B_{107,698}$	…	$B_{107,707}$	$B_{107,715}$

FIG. 10B

172B					8B				
$B_{0,43}$	$B_{0,245}$	···	$B_{0,429}$	$B_{0,601}$	$B_{0,690}$	$B_{0,698}$	···	$B_{0,707}$	$B_{0,715}$
$B_{1,43}$	$B_{1,245}$	···	$B_{1,429}$	$B_{1,601}$	$B_{1,690}$	$B_{1,698}$	···	$B_{1,707}$	$B_{1,715}$
⋮	⋮	···	⋮	⋮	⋮	⋮	···	⋮	⋮
$B_{5,43}$	$B_{5,245}$	···	$B_{5,429}$	$B_{5,601}$	$B_{5,690}$	$B_{5,698}$	···	$B_{5,707}$	$B_{5,715}$
$B_{96,43}$	$B_{96,245}$	···	$B_{96,429}$	$B_{96,601}$	$B_{96,690}$	$B_{96,698}$	···	$B_{96,707}$	$B_{96,715}$
$B_{5,43}$	$B_{6,245}$	···	$B_{6,429}$	$B_{6,601}$	$B_{6,690}$	$B_{6,698}$	···	$B_{6,707}$	$B_{6,715}$
$B_{7,43}$	$B_{7,245}$	···	$B_{7,429}$	$B_{7,601}$	$B_{7,690}$	$B_{7,698}$	···	$B_{7,707}$	$B_{7,715}$
⋮	⋮	···	⋮	⋮	⋮	⋮	···	⋮	⋮
$B_{11,43}$	$B_{11,245}$	···	$B_{11,429}$	$B_{11,601}$	$B_{11,690}$	$B_{11,698}$	···	$B_{11,707}$	$B_{11,715}$
$B_{97,0}$	$B_{97,172}$	···	$B_{97,366}$	$B_{97,558}$	$B_{97,688}$	$B_{97,696}$	···	$B_{97,705}$	$B_{97,713}$
⋮	⋮	···	⋮	⋮	⋮	⋮	···	⋮	⋮
$B_{90,43}$	$B_{90,245}$	···	$B_{90,429}$	$B_{90,601}$	$B_{90,690}$	$B_{90,698}$	···	$B_{90,707}$	$B_{90,715}$
$B_{91,43}$	$B_{91,245}$	···	$B_{91,429}$	$B_{91,601}$	$B_{91,690}$	$B_{91,698}$	···	$B_{91,707}$	$B_{91,715}$
⋮	⋮	···	⋮	⋮	⋮	⋮	···	⋮	⋮
$B_{95,43}$	$B_{95,245}$	···	$B_{95,429}$	$B_{95,601}$	$B_{95,690}$	$B_{95,698}$	···	$B_{95,707}$	$B_{95,715}$
$B_{107,86}$	$B_{107,258}$	···	$B_{107,472}$	$B_{107,644}$	$B_{107,692}$	$B_{107,700}$	···	$B_{107,709}$	$B_{107,717}$

FIG. 10C

1728					88				
$B_{0,86}$	$B_{0,258}$	⋯	$B_{0,472}$	$B_{0,644}$	$B_{0,692}$	$B_{0,700}$	⋯	$B_{0,709}$	$B_{0,717}$
$B_{1,86}$	$B_{1,258}$	⋯	$B_{1,472}$	$B_{1,644}$	$B_{1,692}$	$B_{1,700}$	⋯	$B_{1,709}$	$B_{1,717}$
⋮	⋮	⋯	⋮	⋮	⋮	⋮	⋯	⋮	⋮
$B_{5,86}$	$B_{5,258}$	⋯	$B_{5,472}$	$B_{5,644}$	$B_{5,692}$	$B_{5,700}$	⋯	$B_{5,709}$	$B_{5,717}$
$B_{96,86}$	$B_{96,258}$	⋯	$B_{96,472}$	$B_{96,644}$	$B_{96,692}$	$B_{96,700}$	⋯	$B_{96,709}$	$B_{96,717}$
$B_{6,86}$	$B_{6,258}$	⋯	$B_{6,472}$	$B_{6,644}$	$B_{6,692}$	$B_{6,700}$	⋯	$B_{6,709}$	$B_{6,717}$
$B_{7,86}$	$B_{7,258}$	⋯	$B_{7,472}$	$B_{7,644}$	$B_{7,692}$	$B_{7,700}$	⋯	$B_{7,709}$	$B_{7,717}$
⋮	⋮	⋯	⋮	⋮	⋮	⋮	⋯	⋮	⋮
$B_{11,86}$	$B_{11,258}$	⋯	$B_{11,472}$	$B_{11,644}$	$B_{11,692}$	$B_{11,700}$	⋯	$B_{11,709}$	$B_{11,717}$
$B_{97,43}$	$B_{97,215}$	⋯	$B_{97,429}$	$B_{97,601}$	$B_{97,690}$	$B_{97,698}$	⋯	$B_{97,707}$	$B_{97,715}$
⋮	⋮	⋯	⋮	⋮	⋮	⋮	⋯	⋮	⋮
$B_{90,86}$	$B_{90,258}$	⋯	$B_{90,472}$	$B_{90,644}$	$B_{90,692}$	$B_{90,700}$	⋯	$B_{90,709}$	$B_{90,717}$
$B_{91,86}$	$B_{91,258}$	⋯	$B_{91,472}$	$B_{91,644}$	$B_{91,692}$	$B_{91,700}$	⋯	$B_{91,709}$	$B_{91,717}$
⋮	⋮	⋯	⋮	⋮	⋮	⋮	⋯	⋮	⋮
$B_{95,86}$	$B_{95,258}$	⋯	$B_{95,472}$	$B_{95,644}$	$B_{95,692}$	$B_{95,700}$	⋯	$B_{95,709}$	$B_{95,717}$
$B_{107,129}$	$B_{107,301}$	⋯	$B_{107,515}$	$B_{107,687}$	$B_{107,694}$	$B_{107,702}$	⋯	$B_{107,711}$	$B_{107,719}$

FIG. 10D

172B					8B				
$B_{0,129}$	$B_{0,301}$	···	$B_{0,515}$	$B_{0,687}$	$B_{0,694}$	$B_{0,702}$	···	$B_{0,711}$	$B_{0,719}$
$B_{1,129}$	$B_{1,301}$	···	$B_{1,515}$	$B_{1,687}$	$B_{1,694}$	$B_{1,702}$	···	$B_{1,711}$	$B_{1,719}$
⋮	⋮	···	⋮	⋮	⋮	⋮	···	⋮	⋮
$B_{5,129}$	$B_{5,301}$	···	$B_{5,515}$	$B_{5,687}$	$B_{5,694}$	$B_{5,702}$	···	$B_{5,711}$	$B_{5,719}$
$B_{96,129}$	$B_{96,301}$	···							
$B_{6,129}$	$B_{6,301}$	···	$B_{6,515}$	$B_{6,687}$	$B_{6,694}$	$B_{6,702}$	···	$B_{6,711}$	$B_{6,719}$
$B_{7,129}$	$B_{7,301}$	···	$B_{7,515}$	$B_{7,687}$	$B_{7,694}$	$B_{7,702}$	···	$B_{7,711}$	$B_{7,719}$
⋮	⋮	···	⋮	⋮	⋮	⋮	···	⋮	⋮
$B_{11,129}$	$B_{11,301}$	···	$B_{11,515}$	$B_{11,687}$	$B_{11,694}$	$B_{11,702}$	···	$B_{11,711}$	$B_{11,719}$
		···					···		
⋮	⋮	···	⋮	⋮	⋮	⋮	···	⋮	⋮
$B_{90,129}$	$B_{90,301}$	···	$B_{90,515}$	$B_{90,687}$	$B_{90,694}$	$B_{90,702}$	···	$B_{90,711}$	$B_{90,719}$
$B_{91,129}$	$B_{91,301}$	···	$B_{91,515}$	$B_{91,687}$	$B_{91,694}$	$B_{91,702}$	···	$B_{91,711}$	$B_{91,719}$
⋮	⋮	···	⋮	⋮	⋮	⋮	···	⋮	⋮
$B_{95,129}$	$B_{95,301}$	···	$B_{95,515}$	$B_{95,687}$	$B_{95,694}$	$B_{95,702}$	···	$B_{95,711}$	$B_{95,719}$
		···					···		